

CS 305: SOFTWARE ENGINEERING (3-0-2: 4)

Introduction

Software process - software development life cycle models.

Software Requirement and Analysis

Techniques: feasibility analysis, requirements elicitation, validation, rapid prototyping, OO paradigms vs. structured paradigm - OO analysis.

Software Specifications

Specification document, specification qualities, uses, system modeling: context, interaction, structural, behavioral, DFD, specification techniques using UML, ER diagrams, logic, algebraic specifications: comparison of various techniques, formal specifications – model checking, introduction to binary decision diagrams.

Object Oriented Methodology

Introduction to objects, relationships, unified approach to modeling, use-case modeling, activity, state and interaction diagrams, classification approaches, cohesion, coupling, reuse, case studies - object oriented paradigm, software design: architectural - distributed - data oriented design & object oriented design - real-time systems design techniques.

Stepwise Refinement

Stepwise refinement, software versions and configuration control.

Software Testing & Evolution

Verification & validation – non-execution based testing – software inspections, code reviews, code walkthroughs – automated static analysis – Clean room software development – quality issues – execution based testing – module test-case selection, testing process: black-box, white-box, unit, integration.

Suggested list of Experiments:

- 01) Lab assignments on static program verification tool (SLAM) for verifying critical program behavior.
- 02) Software requirements analysis, design, implementation and testing of the mini-projects assigned in the class (to be done necessarily group-wise in groups of at least two/three students).
- 03) Formal verification of concurrent systems using SPIN model checker.

Text Books

1. Roger S Pressman: "Software Engineering – A Practitioner's Approach", McGraw-Hill.

References

2. Ian Sommerville: "Software Engineering".
 3. SLAM Reference-<http://research.microsoft.com/en-us/projects/slam/>
 4. SPIN Model Checker Reference: <http://spinroot.com/spin/whatispin.html>
-